PÁZMÁNY
Pázmány Péter Catholic University
Faculty of Information Technology and Bionics

Parameter Estimation

Modeling of stock prices

Csanad Tabajdi, Garo Kechichian
Computer Science Engineering MSc

2024

Task Description: Stock Price Prediction Using Differential Equation Modeling

Objective:

The objective of this task is to develop a mathematical model for predicting stock prices based on a differential equation framework. The model will incorporate stochastic processes to account for market uncertainties and will be evaluated using historical stock price data.

Model Description:

Consider the differential equation:

$$dS(t) = \mu_1 S(t)\,dt + \mu_2 S(t)^2\,dt + \ldots + \mu_k S(t)^k\,dt + \sigma S(t)\,dz$$

where:

- $S(t) \in R$ represents the stock price signal.
- $\mu_1, \mu_2, \ldots, \mu_k, \sigma \in R$ are parameters to be estimated.
- dz is a stochastic process (Wiener process) with independent white noise increments $dz \sim N(0, 1)$.

Task Requirements:

1. **Data Collection and Preparation:**
    - Gather historical stock price time series data suitable for the model development.
    - Split the dataset into training and testing sets to evaluate model performance.
2. **Model Fitting:**
    - Fit the differential equation model to the training data using different values of k.
    - Explore models with varying complexities to determine the optimal k.
3. **Model Evaluation:**
    - Evaluate the predictive performance of each model on the test set.

- Measure accuracy metrics such as Mean Squared Error (MSE) or Root Mean Squared Error (RMSE) to assess model effectiveness.

4. **Consideration of Time-Varying Parameters:**

   - Extend the modeling approach to accommodate time-varying parameters.
   - Investigate methods to adapt the model as market conditions change over time.

Remarks:

- The specific value of k is unknown and should be determined through experimentation.
- The term $\sigma S(t)\,dz$ should be considered as an integral part of the modeling approach, rather than a measurement noise.
- Incorporate advanced techniques to handle time-varying parameters for enhanced predictive capabilities.

Generate Dataset

The first step in this task involved preparing a dataset for parameter estimation using historical stock price data. The dataset was obtained from the Kaggle dataset repository, specifically from the 'All Stocks 5yr' dataset available at `https://www.kaggle.com/datasets/rohitjain454/all-stocks-5yr/data`.

Dataset Details:

The dataset comprises daily closing prices ('close' prices) of various stocks over a period of five years. For the purpose of this task, the 'close' prices were used as the modeling signal.

Data Preparation:

- The dataset was split into training and testing sets using an 80-20 split ratio.
- The training set was used for model training and parameter estimation, while the testing set was used for model evaluation and validation.

Modeling Approach:

The task involved fitting a polynomial regression model to predict future stock prices. The model was constructed with varying degrees (k) to determine the optimal degree that minimizes prediction error.

Parameter Estimation Methods

In this section, we discuss the parameter estimation methods utilized in the stock price prediction task, namely Least Squares (LSQ), Recursive Least Squares (RLSQ), and Gradient Descent.

Least Squares (LSQ) Estimator

Context: LSQ is a fundamental method for estimating parameters in linear regression models. It seeks to minimize the sum of squared differences between observed and predicted values.

Explanation: In the context of predicting stock prices based on polynomial features, LSQ solves the normal equations:

$$X^\top X\theta = X^\top Y$$

where:

- X is the design matrix constructed from polynomial features.
- Y is the vector of target values (stock prices).
- θ are the coefficients to be estimated.

Regularization with parameter λ is applied to control overfitting.

Comparison: LSQ provides an efficient solution for parameter estimation in linear models but assumes a linear relationship between predictors and the response variable.

Recursive Least Squares (RLSQ)

Context: RLSQ extends LSQ to adaptively estimate parameters over time, suitable for scenarios with streaming data or evolving models.

Explanation: RLSQ updates parameter estimates iteratively using the Kalman gain K_t and covariance matrix P:

$$\theta_t = \theta_{t-1} + K_t \left(Y_t - X_t^\top \theta_{t-1} \right)$$

$$P_t = \frac{1}{\lambda + X_t^\top P_{t-1} X_t} P_{t-1}$$

where X_t and Y_t are the current observation and target, respectively.

Comparison: RLSQ adapts to changing data patterns and is effective for non-stationary processes but requires parameter tuning and computational resources.

Gradient Descent

Context: Gradient Descent iteratively optimizes model parameters by descending along the gradient of a cost function.

Explanation: In the stock price prediction task, Gradient Descent updates parameters θ as follows:

$$\theta := \theta - \alpha \nabla J(\theta)$$

where α is the learning rate and $\nabla J(\theta)$ is the gradient of the cost function $J(\theta)$.

Comparison: Gradient Descent is versatile for optimizing non-convex functions and handles large datasets efficiently but requires careful tuning of hyperparameters and may converge to local minima.

Model Performance Analysis

In this section, we analyze the performance of different polynomial models for predicting stock prices. The constant k, representing the degree of the polynomial, is not known a priori and must be determined through experimentation. We evaluate models with polynomial degrees ranging from 1 to $k_{\max}$.

0.0.1 Choosing the Best Model

We iteratively constructed features for polynomial degrees $k = 1, 2, 3, 4$. For each degree, we trained models using three different methods: Least Squares (LSQ), Recursive Least Squares (RLSQ), and Gradient Descent (GD). The Mean Squared Error (MSE) was computed for each model to determine the best performing polynomial degree. 1

Figure 1 shows the cost associated with each polynomial degree for the three methods. It is evident that as the degree k increases, the model becomes more accurate. However, higher polynomial degrees can lead to overfitting, where the model captures noise in the training data rather than the underlying trend.

After evaluating the costs, we found that the best model has a polynomial degree $k = 4$ with the following performance metrics on the test data:

- **MSE:** 3.4535
- R^2**:** 0.9983

0.0.2 Simulated Stock Prices

Using the best model parameters, we simulated stock prices for various scenarios: single time step, daily, and weekly. The performance metrics for these simulations are summarized below:

- **Simulated:**
  - **MSE:** 39122558.6296
  - R^2**:** -18757.2490
- **Daily:**
  - **MSE:** 1695.5559
  - R^2**:** 0.1870
- **Weekly:**
  - **MSE:** 1739.7977
  - R^2**:** 0.1658

Figures 2, 3, and 4 illustrate the actual and simulated stock prices for the best model, daily simulation, and weekly simulation, respectively.

0.0.3 Sensitivity Analysis

To understand the robustness of our model, we performed a sensitivity analysis by varying the model parameters (θ and σ) by $\pm 10\%$. The results are shown in Figures 5 and 6.

The sensitivity analysis reveals that variations in the model parameters can lead to better performance in some cases. This indicates that the model is somewhat robust but can be further fine-tuned for improved accuracy.

Conclusion

In conclusion, the polynomial regression model with $k = 4$ provides an excellent fit to historical stock price data, achieving high accuracy as indicated by low MSE and high R-squared values during testing. The model's ability to simulate future prices however, caution is advised when extrapolating these results over longer periods, as evidenced by the divergence in simulation metrics. Also to achieve better results, we need to improve the stochastic process inside the model, but we can get valuable results for shorter periods. While increasing the polynomial degree generally improves model accuracy, it also risks overfitting. The sensitivity analysis suggests that the model parameters can be adjusted to potentially enhance performance. Future enhancements could explore more robust methodologies or incorporate additional features to improve the model's generalizability and accuracy in long-term forecasting scenarios.

0.1. figures

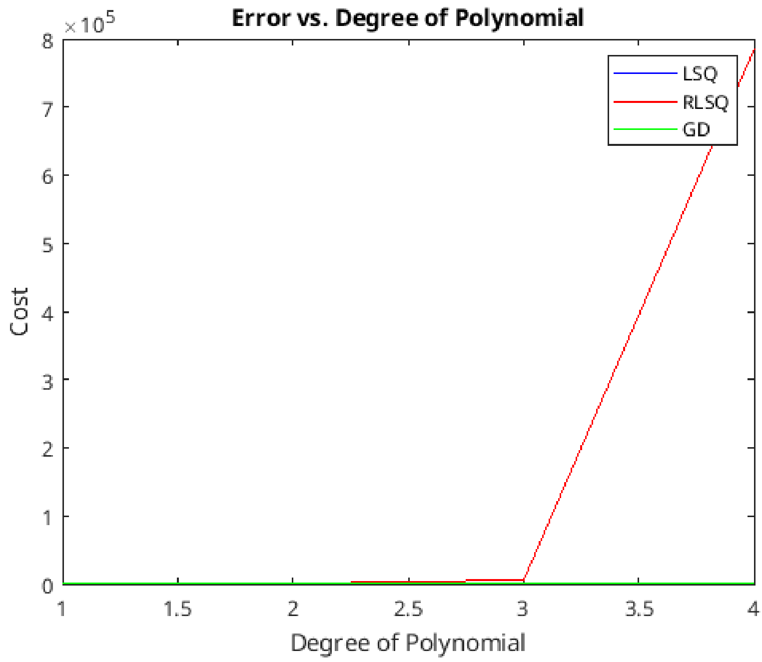

Figure 1: Error vs. Degree of Polynomial

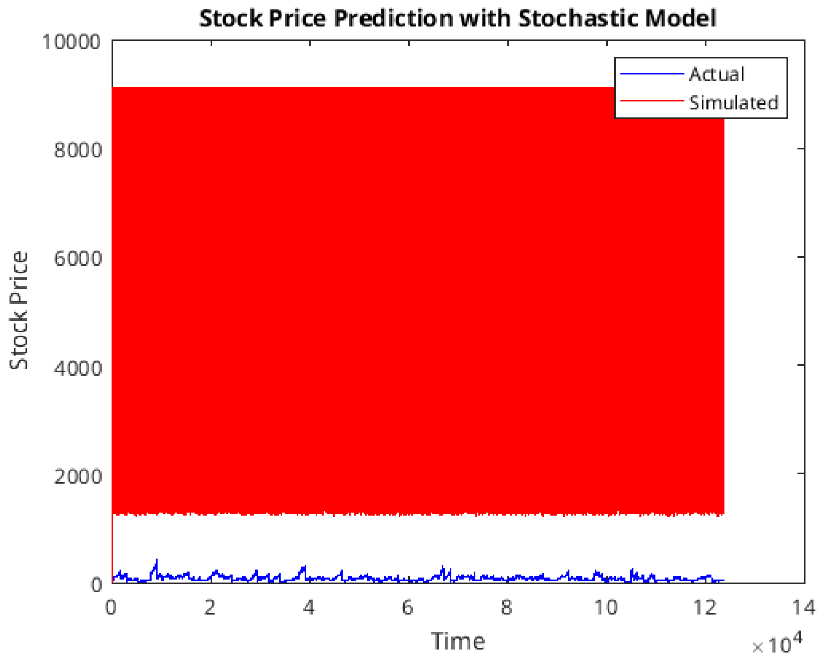

Figure 2: Stock Price Prediction with Best Model

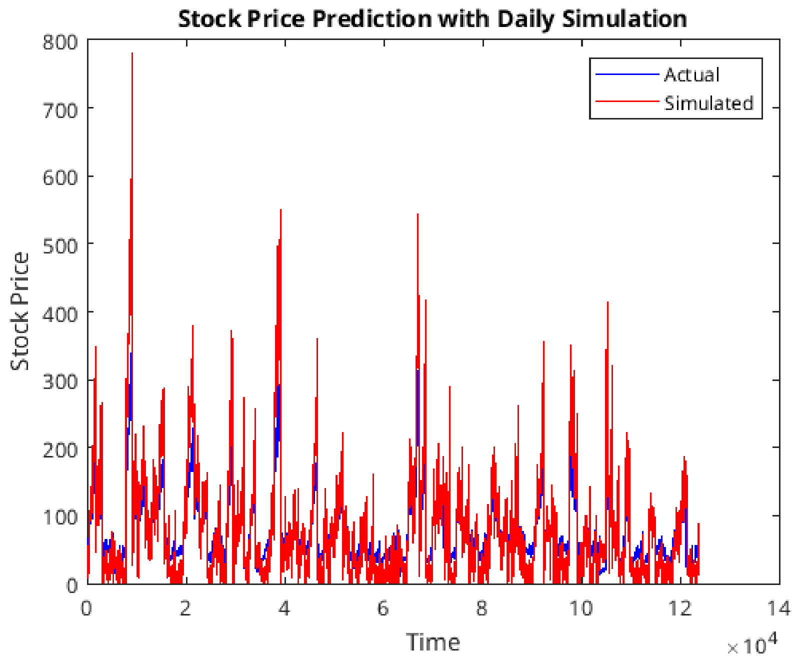

Figure 3: Stock Price Prediction with Daily Simulation

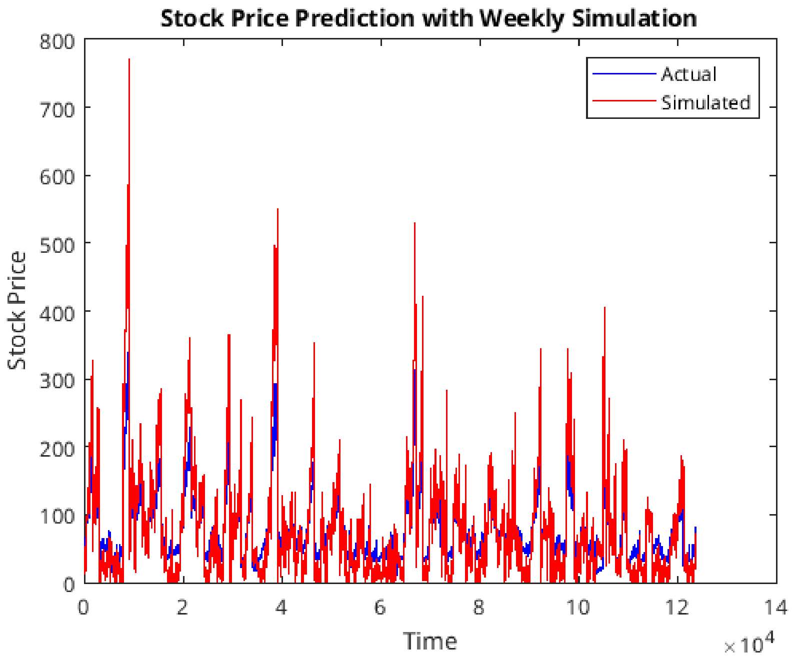

Figure 4: Stock Price Prediction with Weekly Simulation

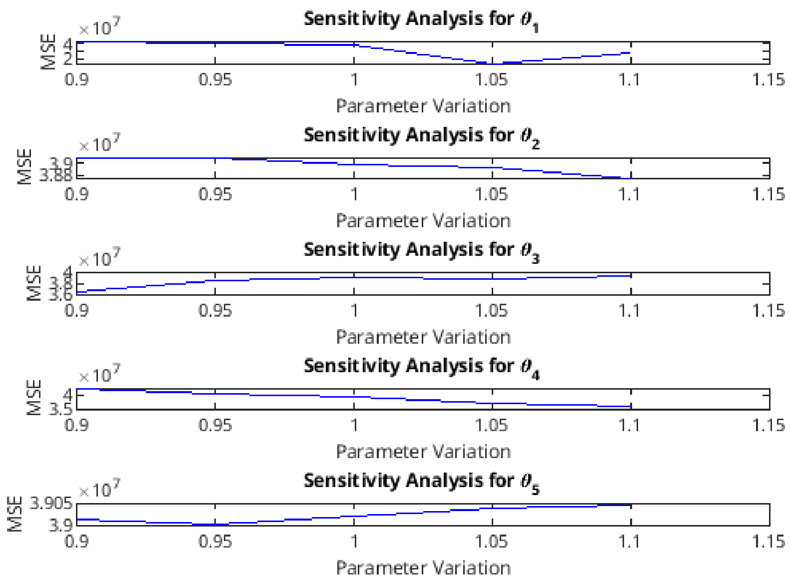

Figure 5: Sensitivity Analysis for θ

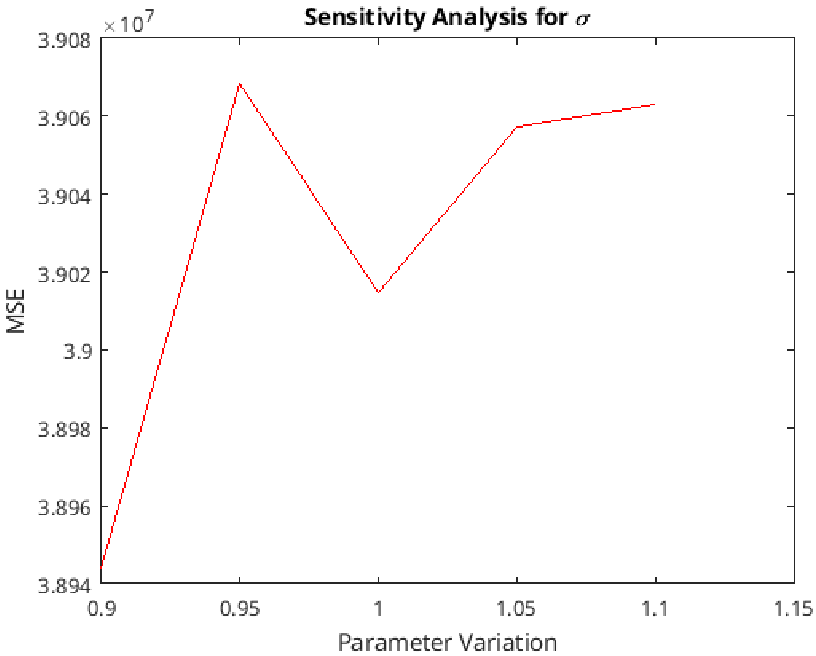

Figure 6: Sensitivity Analysis for σ